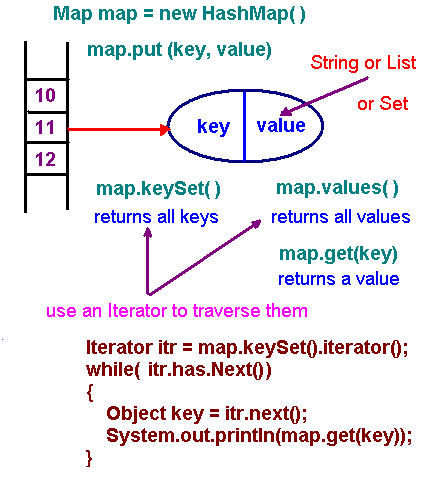
# Datové struktury – semestrální práce A

Naším úkolem bylo:

Z jednoho výchozího uzlu (obce) do všech ostatních dosažitelných uzlů, resp. ze všech uzlů sítě do jednoho koncového uzlu.

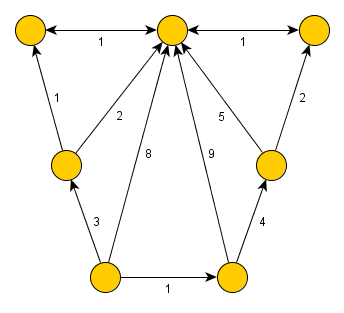
## Datová struktura :

Pro reprezentaci vrcholů a hran jsem si vybral datovou strukturu HashMap, kterou nabízí jazyk JAVA v základních kolekcích, do této datové struktury jsem ukládal generický klíč a vrcholy. Tuto strukturu jsem si vybral na základě efektivního vyhledávání klíče. Klíč musí být jedinečný nelze vložit duplicitní klíč. Jelikož vkládání a odebírání se v dané semestrální práci využívá jen výjimečně tak jsem neřešil složitost vkládání a odebírání. Kdy se musí projít celá daná struktura. Při vložení stejného klíče se nám starý klíč změní za nový klíč.



## Algoritmus pro vyhledávání : Dijkstrův algoritmus

Tento algoritmus jsem si vybral že se jedná o nejrychlejší algoritmus pro nalezení všech nejkratších cest ze zadaného uzlu do ostatních uzlů. Složitost Dijkstrova algoritmu závisí na implementaci prioritní fronty. V případě její implementace pomocí sekvenčního vyhledávání je složitost algoritmu O(\vert U \vert ^{2}). Tento algoritmus jsem postavil na vkládání a odebírání vrcholů z prioritní fronty.



Dále jsem v této práci používal další kolekce jako jsou LinkendList, Set.

## Vrcholy a hrany – třída Graf

Uchovávání vrcholů jak jsem již psal je ukládáno do datové struktury hashmap. Dále hrany jsou ukládány do LinkedListu. Každý vrchol má svůj seznam hran. Tím jsem docílil že datové struktura graf je neorientovaný graf. Jelikož privátní třída hrana obsahuje referenci odkud a kam jde z jakého města do cílového města. Všechna vkládání a odebírání jsou generická zde se dají uložit jaká koliv data.

## Města a silnice – třída Mapa

Zde se již nacházíme nad datovou částí kde vkládáme města a vrcholy do třídy map kde ji také využíváme. Tuto třídu využíváme i k vypočítání nejkratších cest. Kde využívám Dijkstrův algoritmus.

## Třída main

Tato třída slouží ke spouštění programu.